



警示

1. 实验报告如有雷同，雷同各方当次实验成绩均以 0 分计。
2. 当次小组成员成绩只计学号、姓名登录在下表中的。
3. 在规定时间内未上交实验报告的，不得以其他方式补交，当次成绩按 0 分计。
4. 实验报告文件以 PDF 格式提交。

院系	数据科学与计算机学院	班 级	周四三四节	组长	王晶
学号	16340217	16340319	16340205		
学生	王晶	庄文梓	汤万鹏		
实验分工					
王晶	负责实验和实验报告		庄文梓	负责实验和实验报告	
汤万鹏	查找资料				

【实验题目】端口聚合实验

【实验目的】理解链路聚合的配置及原理。

【实验内容】

(1)

第二版：

完成实验教程 6-5 端口聚合配置实验，回答实验提出的问题及实验思考。(P187)

第一版：

完成实验教程 3-5 端口聚合配置实验，回答实验提出的问题及实验思考。(P99)

(2) 端口聚合和生成树都可以实现冗余链路，这两种方式有什么不同？

(3) 对本实验的内容，你们小组还能提出其他更难的实验要求吗？如能，请设计 1-2 个，并给出解答。

【实验要求】

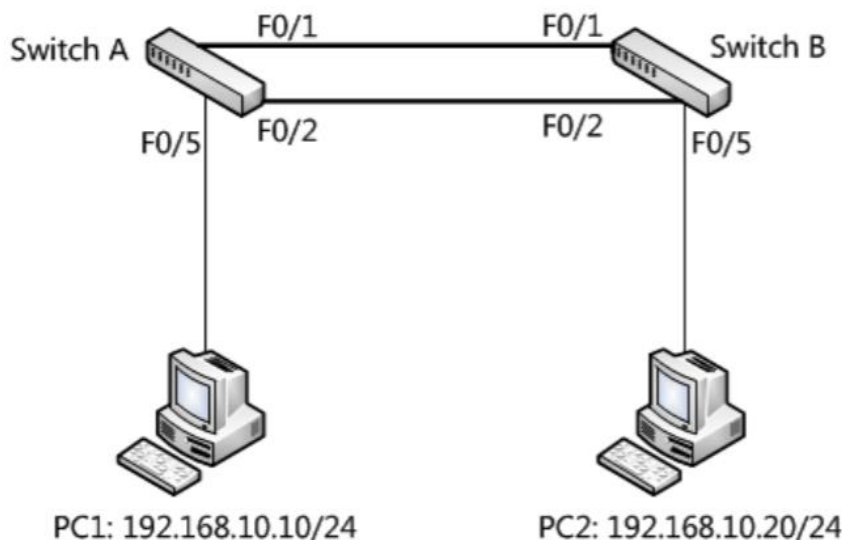
一些重要信息需给出截图。

注意实验步骤的前后对比！

【实验记录】(如有实验拓扑请自行画出，要求自行画出拓扑图)

(1)

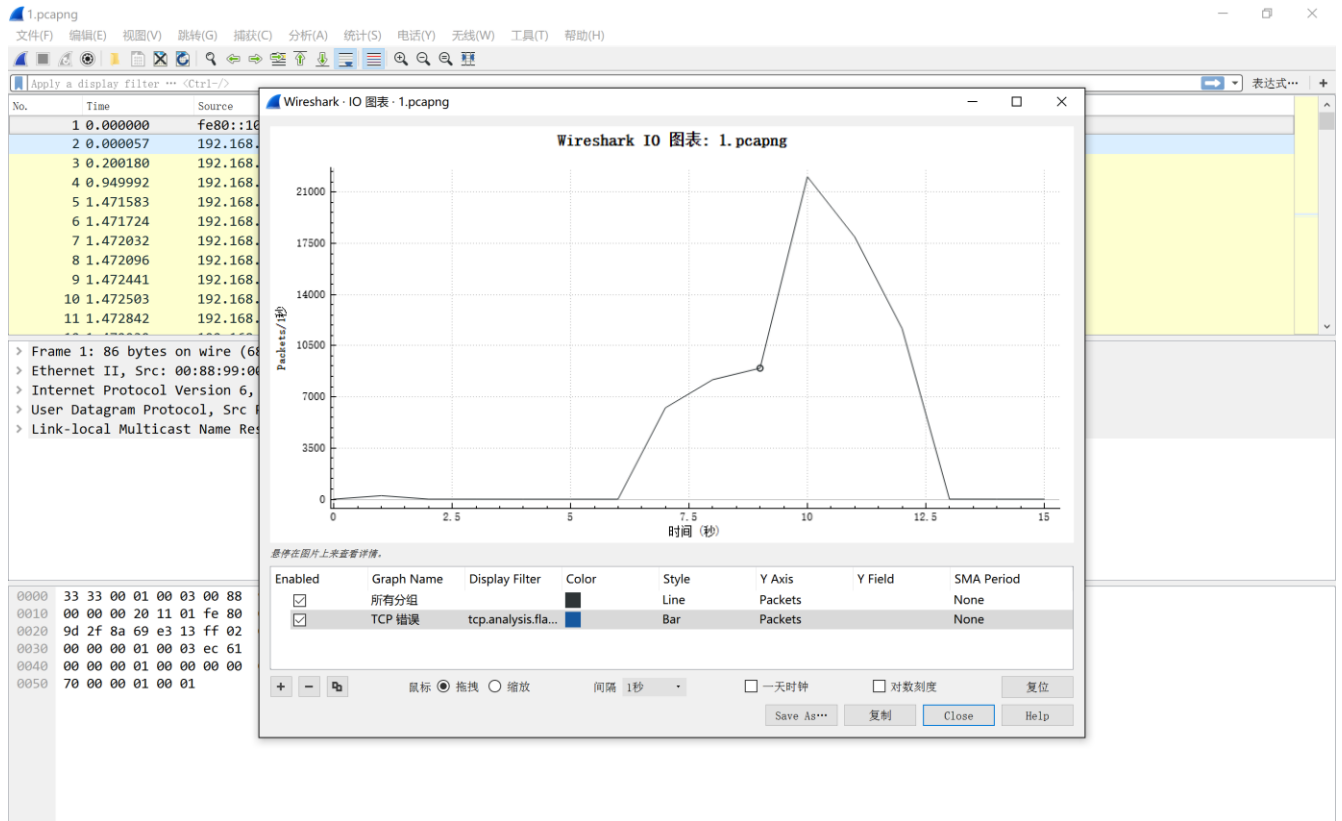
拓扑图：





计算机网络实验报告

按照拓扑图连好网线，在 PC2 上建立一个共享目录，然后启动抓包软件，选中监控对象，界面停留在 capture interface:



可以看到一开始 PC 之间没有数据传输，传输速度是 0。

然后开始进行文件传输，并记录下了传输速率的变化。

现在交换机 A 上配置聚合端口，完成交换机 A 的配置后，验证 F0/1, F0/2 是否属于 AG1:

```
19-S5750-1#show aggregatePort 1 summary
AggregatePort MaxPorts SwitchPort Mode Ports
-----
Ag1            8      Enabled   TRUNK Gi0/1 ,Gi0/2
```

可以看到，对于聚合端口 AG1，其最大支持的端口数为 8。当前 VLAN 模式为 Trunk 模式。且端口 0/1 和 0/2 都属于 AG1。因此，聚合端口 AG1 的配置是正确的。

再验证已经在交换机 B 上创建了 VLAN10，并将 F0/5 划分到 VLAN10 中:

```
19-S5750-2(config-if-GigabitEthernet 0/5)#show vlan id 10
VLAN Name                Status    Ports
-----
10 sales                  STATIC    Gi0/5
```

然后在交换机 B 上配置聚合端口，验证 F0/1, F0/2 是否属于 AG1:

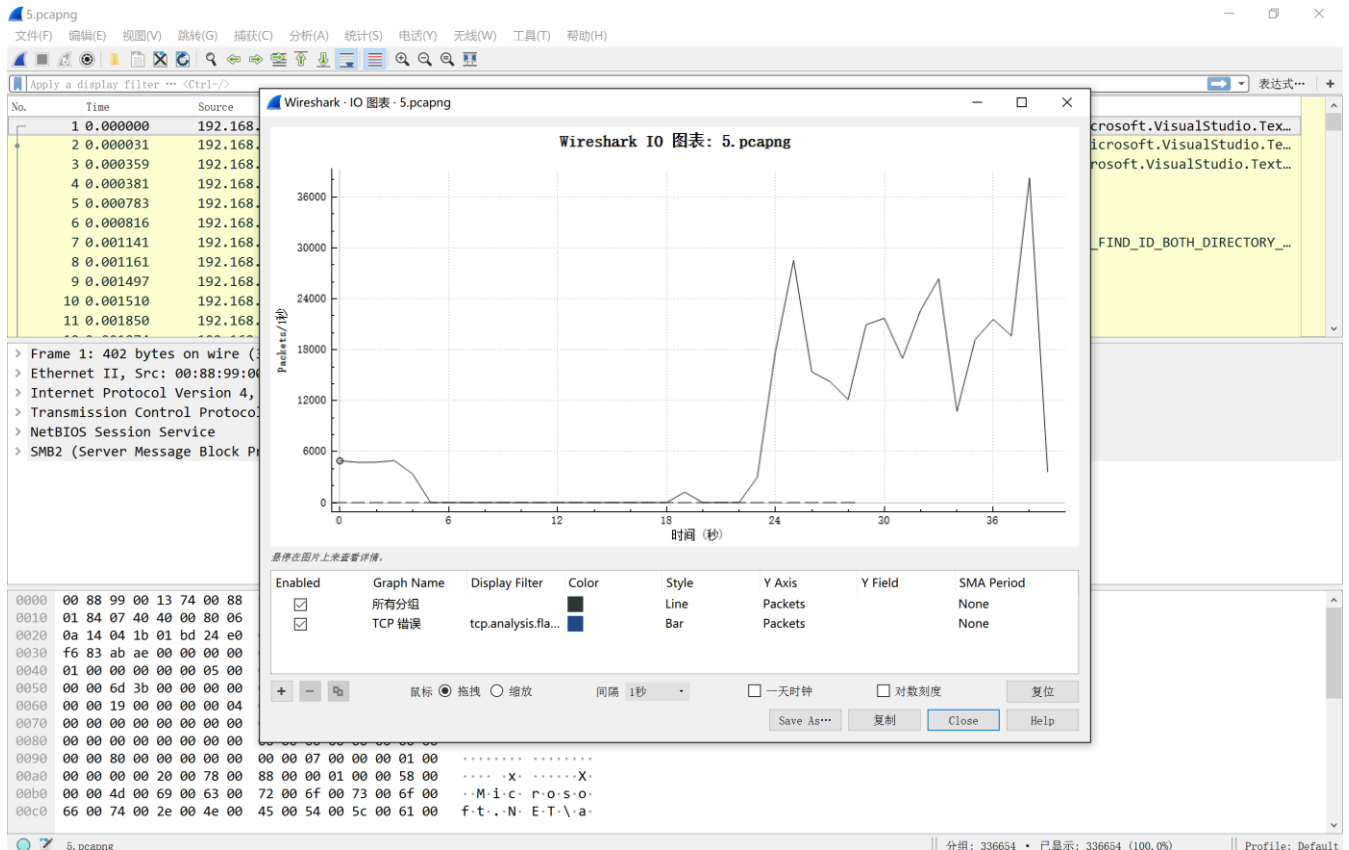
```
19-S5750-2#show aggregatePort 1 summary
AggregatePort MaxPorts SwitchPort Mode Ports
-----
Ag1            8      Enabled   TRUNK Gi0/1 ,Gi0/2
```



计算机网络实验报告

可以看到，对于聚合端口 AG1，其最大支持的端口数为 8。当前 VLAN 模式为 Trunk 模式。且端口 0/1 和 0/2 都属于 AG1。因此，聚合端口 AG1 的配置是正确的。

验证链路聚合的带宽是否增大，如没有，分析原因，并解决。



重新进行传输，并记录传输速率，发现和之前的相比，并没有什么太大的变化。相对于只使用一根线的情况，传输速率并没有什么大增长。不过总的来看，是略有提高，也产生了波动。

这可能来自于两个原因：

A. 带宽瓶颈在 PC 与交换机之间，而非两台交换机之间

从实验拓扑图可以看出，PC 与交换机的连线始终只有一条。因此，就算两台交换机之间只连接一条线，交换机之间的传输带宽也能满足 PC 与交换机之间的传输速率。

B. 流量平衡模式不会将 PC1 向 PC2 的数据传输分流到两条线上

两台交换机在默认情况下的流量平衡模式是源 MAC+目的 MAC。当 PC1 向 PC2 发送数据包的时候，由于所有传输报文的源 MAC 都是 PC1 的 MAC，所有传输报文的目的 MAC 都是 PC2 的 MAC，所有所有的报文都将经过相同的成员链路进行转发。因此即便交换机连上了两条线，PC1 向 PC2 发送的数据也只会沿着其中一条线发送。

链路聚合的动态备份，当交换机之间的一条链路断开时，PC1 与 PC2 仍能互相通信：在 PC1 下一直 ping PC2。过程中将 2 条跳线中 1 条拔掉，观察。



```
C:\Users\Administrator>ping 192.168.10.20 -t

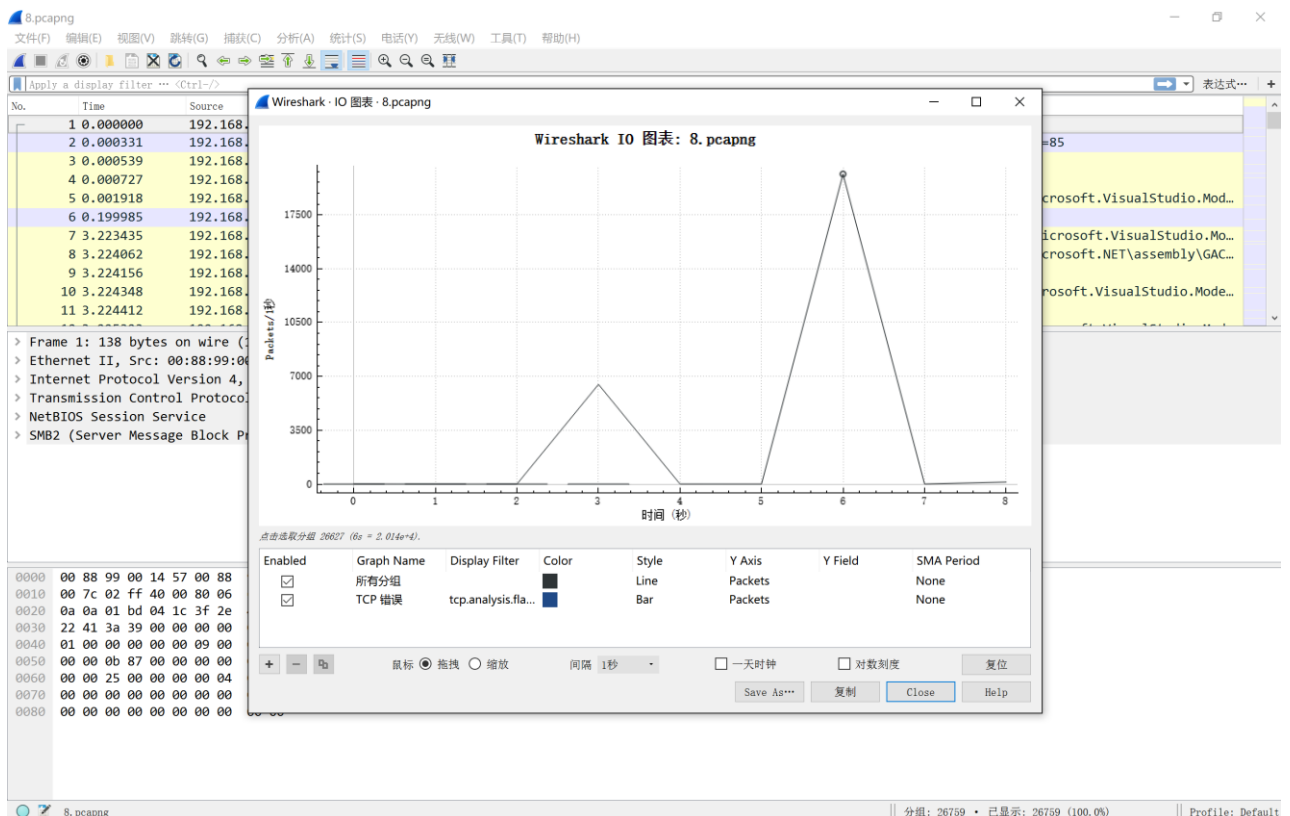
正在 Ping 192.168.10.20 具有 32 字节的数据:
来自 192.168.10.20 的回复: 字节=32 时间=4ms TTL=128
来自 192.168.10.20 的回复: 字节=32 时间=1ms TTL=128
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=128
来自 192.168.10.20 的回复: 字节=32 时间=1ms TTL=128
来自 192.168.10.20 的回复: 字节=32 时间=1ms TTL=128
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=128
来自 192.168.10.20 的回复: 字节=32 时间=1ms TTL=128
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=128

192.168.10.20 的 Ping 统计信息:
    数据包: 已发送 = 8, 已接收 = 8, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 0ms, 最长 = 4ms, 平均 = 1ms
Control-C
^C
C:\Users\Administrator>
```

可以看到，仍能正常 ping 通。

端口聚合本质上是将多个端口聚合在一起形成 1 个汇聚组。端口汇聚服务的上层实体把同一聚合组内的多条物理链路视为一条逻辑链路，从整体上看，1 个汇聚组好像就是 1 个端口。通过端口聚合，多个端口会被汇聚为一组，并且同组成员彼此动态备份。因此，当采用端口聚合后，即使拔掉了一根线，另一根线作为拔掉的线的冗余备份，数据仍可以通过这根线进行传输，网络依旧畅通。

重做步骤 5 验证 1，并在过程中拔掉一根线，观察：





有变化，传输速率出现了变化。

查看聚合口：

```
19-S5750-1(config)#show interfaces aggregateport 1
Index(dec):29 (hex):1d
AggregatePort 1 is UP , line protocol is UP
Hardware is Aggregate Link AggregatePort
Interface address is: no ip address
MTU 1500 bytes, BW 1000000 Kbit
Encapsulation protocol is Bridge, loopback not set
Keepalive interval is 10 sec , set
Carrier delay is 2 sec
Rxload is 1/255, Txload is 1/255
Switchport attributes:
  interface's description: ""
  medium-type is copper
  lastchange time:0 Day: 0 Hour:17 Minute: 6 Second
  Priority is 0
  admin duplex mode is AUTO, oper duplex is Full
  admin speed is AUTO, oper speed is 1000M
  flow control admin status is OFF, flow control oper status is OFF
  Storm Control: Broadcast is ON, Multicast is OFF, Unicast is ON
Aggregate Port Informations:
  Aggregate Number: 1
  Name: "AggregatePort 1"
  Refs: 2
  Members: (count=2)
    GigabitEthernet 0/1      Link Status: Up
    GigabitEthernet 0/2      Link Status: Down
```

查看成员口：

```
19-S5750-1(config)#show interfaces gigabitEthernet 0/1
Index(dec):1 (hex):1
GigabitEthernet 0/1 is administratively down , line protocol is DOWN
Hardware is Broadcom 5464 GigabitEthernet
Interface address is: no ip address
MTU 1500 bytes, BW 1000000 Kbit
Encapsulation protocol is Bridge, loopback not set
Keepalive interval is 10 sec , set
Carrier delay is 2 sec
Rxload is 1/255, Txload is 1/255
Switchport attributes:
  interface's description: ""
  medium-type is copper
  lastchange time:0 Day: 0 Hour: 0 Minute:53 Second
  Priority is 0
  admin duplex mode is AUTO, oper duplex is Full
  admin speed is AUTO, oper speed is 1000M
  flow control admin status is OFF, flow control oper status is OFF
  Storm Control: Broadcast is ON, Multicast is OFF, Unicast is ON
5 minutes input rate 35956 bits/sec, 32 packets/sec
5 minutes output rate 2378188 bits/sec, 207 packets/sec
266158 packets input, 75841836 bytes, 0 no buffer, 0 dropped
Received 653 broadcasts, 0 runts, 0 giants
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 abort
1237417 packets output, 1702587451 bytes, 0 underruns , 0 dropped
0 output errors, 0 collisions, 0 interface resets
```



计算机网络实验报告

查看端口状态:

```
19-S5750-1(config)#show interfaces status
```

Interface	Status	Vlan	Duplex	Speed	Type
GigabitEthernet 0/1	up	1	Full	1000M	copper
GigabitEthernet 0/2	down	1	Unknown	Unknown	copper
GigabitEthernet 0/3	down	1	Unknown	Unknown	copper
GigabitEthernet 0/4	down	1	Unknown	Unknown	copper
GigabitEthernet 0/5	up	10	Full	1000M	copper
GigabitEthernet 0/6	down	1	Unknown	Unknown	copper
GigabitEthernet 0/7	down	1	Unknown	Unknown	copper
GigabitEthernet 0/8	down	1	Unknown	Unknown	copper
GigabitEthernet 0/9	down	1	Unknown	Unknown	copper
GigabitEthernet 0/10	down	1	Unknown	Unknown	copper
GigabitEthernet 0/11	down	1	Unknown	Unknown	copper
GigabitEthernet 0/12	down	1	Unknown	Unknown	copper
GigabitEthernet 0/13	down	1	Unknown	Unknown	copper
GigabitEthernet 0/14	down	1	Unknown	Unknown	copper
GigabitEthernet 0/15	down	1	Unknown	Unknown	copper
GigabitEthernet 0/16	down	1	Unknown	Unknown	copper
GigabitEthernet 0/17	down	1	Unknown	Unknown	copper
GigabitEthernet 0/18	down	1	Unknown	Unknown	copper
GigabitEthernet 0/19	down	1	Unknown	Unknown	copper
GigabitEthernet 0/20	down	1	Unknown	Unknown	copper
GigabitEthernet 0/21	down	1	Unknown	Unknown	copper
GigabitEthernet 0/22	down	1	Unknown	Unknown	copper
GigabitEthernet 0/23	down	1	Unknown	Unknown	copper
GigabitEthernet 0/24	down	1	Unknown	Unknown	copper
GigabitEthernet 0/25	down	1	Unknown	Unknown	fiber
GigabitEthernet 0/26	down	1	Unknown	Unknown	fiber
GigabitEthernet 0/27	down	1	Unknown	Unknown	fiber
GigabitEthernet 0/28	down	1	Unknown	Unknown	fiber
AggregatePort 1	up	1	Full	1000M	copper



查看成员口的速率流量:

19-S5750-1(config)#show interfaces counters summary

Interface	InOctets	InUcastPkts	InMulticastPkts	InBroadcastPkts
Gi0/1	75879075	264866	670	679
Gi0/2	45019	0	0	32
Gi0/3	0	0	0	0
Gi0/4	0	0	0	0
Gi0/5	1699473630	1236168	661	712
Gi0/6	0	0	0	0
Gi0/7	0	0	0	0
Gi0/8	0	0	0	0
Gi0/9	0	0	0	0
Gi0/10	0	0	0	0
Gi0/11	0	0	0	0
Gi0/12	0	0	0	0
Gi0/13	0	0	0	0
Gi0/14	0	0	0	0
Gi0/15	0	0	0	0
Gi0/16	0	0	0	0
Gi0/17	0	0	0	0
Gi0/18	0	0	0	0
Gi0/19	0	0	0	0
Gi0/20	0	0	0	0
Gi0/21	0	0	0	0
Gi0/22	0	0	0	0
Gi0/23	0	0	0	0
Gi0/24	0	0	0	0
Gi0/25	0	0	0	0
Gi0/26	0	0	0	0
Gi0/27	0	0	0	0
Gi0/28	0	0	0	0
Ag1	68091286	204075	163	202
Interface	OutOctets	OutUcastPkts	OutMulticastPkts	OutBroadcastPkts
Gi0/1	1702622657	1236168	601	689
Gi0/2	4877	0	35	0
Gi0/3	0	0	0	0
Gi0/4	0	0	0	0
Gi0/5	75060940	264866	566	592
Gi0/6	0	0	0	0
Gi0/7	0	0	0	0
Gi0/8	0	0	0	0
Gi0/9	0	0	0	0
Gi0/10	0	0	0	0
Gi0/11	0	0	0	0
Gi0/12	0	0	0	0
Gi0/13	0	0	0	0
Gi0/14	0	0	0	0
Gi0/15	0	0	0	0
Gi0/16	0	0	0	0
Gi0/17	0	0	0	0
Gi0/18	0	0	0	0
Gi0/19	0	0	0	0
Gi0/20	0	0	0	0
Gi0/21	0	0	0	0
Gi0/22	0	0	0	0



实验思考：

- (1) 设置聚合端口的负载平衡方式命令：

`show aggregateport load-balance`

- (2) 如何验证聚合端口的负载平衡方式。

首先，我们可以先在交换机的配置界面下检验当前交换机是否正确配置了负载平衡。我们可以在交换机的配置界面下，输入“`show aggregateport [port-number] {load-balance | summary }`”命令，查看当前的聚合端口设置。用“`summary`”做参数可以查看指定聚合端口的有关信息（包括最大支持的端口数、VLAN 模式等）；用“`load-balance`”做参数可以查看当前的端口聚合模式。采用这些命令，我们可以初步判断交换机的负载平衡有没有被正确设置。如果理论设置没错，我们则可以进行实验验证。

由于实验交换机有 6 种平衡模式，所以要检验是否正确地实行了负载平衡，我们要针对每种平衡模式采用不同的实验方式。

为了方便说明，我们规定下面所述的网络连接中，PC 机所连接的端口都是在聚合端口的成员组内。并且，在验证过程中，交换机并不传送与验证过程无关的报文。

1) 根据源 MAC 地址进行流量平衡（`src- -c mac` 模式）：我们选择两台具有不同 MAC 地址的 PC 机（PC1 和 PC2），同时向第三台 PC 机（PC3）传送数据。如果此时交换机上有两个端口的指示灯在快速闪动，则说明当前交换机已经将来自 PC1 和 PC2 的报文分配到不同链路上进行传输，负载平衡的设定是正确的。

2) 根据目的 MAC 地址进行流量平衡（`dst- -m mac` 模式）：我们选择一台 PC 机（PC1），同时向其他两台具有不同 MAC 地址的 PC 机（PC2、PC3）传送数据。如果此时交换机上有两个端口的指示灯在快速闪动，则说明当前交换机已经将发往 PC2 和 PC3 的报文分配到不同链路上进行传输，负载平衡的设定是正确的。

3) 根据源 MAC 地址+ + 目的 MAC 地址进行流量平衡（`src- -dst- -c mac` 模式）：我们首先选择两台具有不同 MAC 地址的 PC 机，同时向第三台 PC 机传送数据；接着，我们用一台 PC 机，同时向其他两台具有不同 MAC 地址的 PC 机传输数据。如果在这两次传输数据的过程中，交换机上都有两个端口的指示灯在快速闪动，则说明只要是源 MAC 不同，或者目的 MAC 不同，交换机都会讲报文分配到不同链路上进行传输，负载平衡的设定是正确的。

4) 根据源 IP 地址进行流量平衡（`src- -p ip` 模式）：我们选择两台具有不同 IP 地址的 PC 机（PC1 和 PC2），同时向第三台 PC 机（PC3）传送数据。如果此时交换机上有两个端口的指示灯在快速闪动，则说明当前交换机已经将来自 PC1 和 PC2 的报文分配到不同链路上进行传输，负载平衡的设定是正确的。

5) 根据目的 IP 地址进行流量平衡（`dst- -p ip` 模式）：我们选择一台 PC 机（PC1），同时向其他两台具有不同 IP 地址的 PC 机（PC2、PC3）传送数据。如果此时交换机上有两个端口的指示灯在快速闪动，则说明当前交换机已经将发往 PC2 和 PC3 的报文分配到不同链路上进行传输，负载平衡的设定是正确的。

6) 根据源 IP 地址+目的 IP 地址进行流量平衡（`src- -dst- -p ip` 模式）：我们首先选择两台具有不同 IP 地址的 PC 机，同时向第三台 PC 机传送数据；接着，我们用一台 PC 机，同时向其他两台具有不同 IP 地址的 PC 机传输数据。如果在这两次传输数据的过程中，交换机上都有两个端口的指示灯在快速闪动，则说明只要是源 IP 不同，或者目的 IP 不同，交换机都会讲报文分配到不同链路上进行传输，负载平衡的设定是正确的。



(3) 什么情况下链路聚合会起分流作用。

实验交换机支持 6 种不同的流量平衡模式。使用这 6 种平衡模式，链路聚合进行分流的情形不同。

1) 根据源 MAC 地址进行流量平衡 (src- -c mac 模式): 在这个模式下, 如果报文来自相同的源 MAC, 将经同一个链路进行转发; 只有报文来自不同的源 MAC 时, 报文转发的成员链路才会不同, 即此时链路聚合才会起分流作用。

2) 根据目的 MAC 地址进行流量平衡 (dst- -c mac 模式): 在这个模式下, 如果报文发往相同的目的 MAC, 将经同一个链路进行转发; 只有报文发往不同的目的 MAC 时, 报文转发的成员链路才会不同, 即此时链路聚合才会起分流作用。

3) 根据源 MAC 地址+目的 MAC 地址进行流量平衡 (src- -dst- -c mac 模式): 在这个模式下, 如果报文来自同一个源 MAC, 并且将发往同一个目的 MAC, 则将经同一个链路进行转发; 只有报文来自不同的源 MAC, 或者要发往不同的目的 MAC 时, 报文转发的成员链路才会不同, 即此时链路聚合才会起分流作用。

4) 根据源 IP 地址进行流量平衡 (src- -p ip 模式): 在这个模式下, 如果报文来自相同的源 IP, 将经同一个链路进行转发; 只有报文来自不同的源 IP 时, 报文转发的成员链路才会不同, 即此时链路聚合才会起分流作用。

5) 根据目的 IP 地址进行流量平衡 (dst- -p ip 模式): 在这个模式下, 如果报文发往相同的目的 IP, 将经同一个链路进行转发; 只有报文发往不同的目的 IP 时, 报文转发的成员链路才会不同, 即此时链路聚合才会起分流作用。

6) 根据源 IP 地址+目的 IP 地址进行流量平衡 (src- -dst- -p ip 模式): 在这个模式下, 如果报文来自同一个源 IP, 并且将发往同一个目的 IP, 则将经同一个链路进行转发; 只有报文来自不同的源 IP, 或者要发往不同的目的 IP 时, 报文转发的成员链路才会不同, 即此时链路聚合才会起分流作用。

(2)端口聚合与采用生成树实现冗余链路的不同:

端口聚合 (AP), 又称为链路聚合, 是把交换机多个特性相同的端口物理连接并在逻辑上绑定在一起, 形成一个具有较大宽带的逻辑端口, 将多条物理链路聚合成一条逻辑链路。从而实现均衡负载, 并提供冗余链路。生成树协议, 是在交换网络中通过 SPA(生成树算法)生成一个没有环路的网络。

当主要链路出现故障时, 能够自动重构网络, 将网络连接切换到备份链路, 保证网络的正常通信。生成树协议仅仅是保持一个网络中没有环, 仅起到冗余备份链路的作用, 对网络中的均衡负载没有任何作用。而端口聚合除了起到冗余备份链路作用外, 还可以在 network 各端口上负载分担, 增大链路带宽, 从而解决交换网络中因带宽引起的网络瓶颈问题。但端口聚合的要求比生成树协议更多。端口聚合要求: 只有同类型端口且双工速率一致才能聚合为一个 AG 端口, 光口和电口不能绑定; 所有端口



计算机网络实验报告

必须在同一个 VLAN 下；端口使用的传输介质相同；所有端口与 AP 必须属于同一层次。生成树协议是一个纯二层协议（二层以太网），而链路聚合可以在二层聚合口，也可以在三层聚合口上使用。一般来说，端口聚合技术应用于交换机之间的骨干链路上，或交换机与大流量的服务器之间，以在解决交换网络中环路问题的同时，实现带宽增加和流量平衡。而生成树技术一般用于交换网络，以解决交换网络中的环路问题。

(3)对本实验的内容，你们小组还能提出其他更难的实验要求吗？如能，请设计 1-2 个，并给出解答。



学号	学生	自评分
16340217	王晶	90
16340319	庄文梓	90
16340205	汤万鹏	88

【交实验报告】

上传实验报告：ftp://222.200.181.161

截止日期（不迟于）：1 周之内

上传包括两个文件：

（1）小组实验报告。上传文件名格式：小组号_Ftp 协议分析实验.pdf （由组长负责上传）

例如：文件名“10_Ftp 协议分析实验.pdf”表示第 10 组的 Ftp 协议分析实验报告

（2）小组成员实验体会。每个同学单独交一份只填写了实验体会的实验报告。只需填写自己的学号和姓名。

文件名格式：小组号_学号_姓名_Ftp 协议分析实验.pdf （由组员自行上传）

例如：文件名“10_05373092_张三_Ftp 协议分析实验.pdf”表示第 10 组的 Ftp 协议分析实验报告。

注意：不要打包上传！